

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-143538

(43)Date of publication of application : 29.05.1998

(51)Int.Cl. G06F 17/30
G06F 12/00

(21)Application number : 09-181947 (71)Applicant : INFORMIX SOFTWARE INC

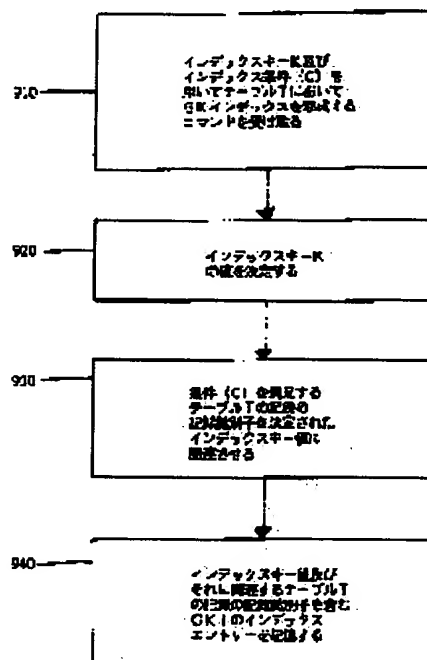
(22)Date of filing : 08.07.1997 (72)Inventor : SUNDARESAN PRAKASH

(30)Priority

Priority number : 96 677490 Priority date : 09.07.1996 Priority country : US

(54) GENERALIZED KEY INDEX

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a new index in a relational data base system.**SOLUTION:** A generalized key index is capable of index the first table of a relational data base through the use of an index key and an index condition. One or both of the index key and the index condition can refer to the many tables of the data base and or else, expression can use the field of at least one table excepting of the first table. The generalized key index directly correlate a generalized index key value with the recording identifier of the recording of the first table to make storage and data retrieval efficient.

LEGAL STATUS

[Date of request for examination] 25.05.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-143538

(43) 公開日 平成10年(1998) 5月29日

(51) Int.Cl.⁹
G 0 6 F 17/30
12/00 5 2 0

F I
G 0 6 F 15/40 3 8 0 D
12/00 5 2 0 A
15/403 3 4 0 D

審査請求 未請求 請求項の数14 O L (全 16 頁)

(21) 出願番号 特願平9-181947

(22) 出願日 平成9年(1997) 7月8日

(31) 優先権主張番号 0 8 / 6 7 7 4 9 0

(32) 優先日 1996年7月9日

(33) 優先権主張国 米国 (U S)

(71) 出願人 596158536

インフォアミックス ソフトウェア イン
コーポレイテッド
アメリカ合衆国 カリフォルニア州
94025 メンロ パーク ボハーノン ド
ライヴ 4100

(72) 発明者 ブラカシュ サンデアサン

アメリカ合衆国 オレゴン州 97205 ポ
ートランド サウスウエスト キング ア
ベニュー 901 アパートメント 710

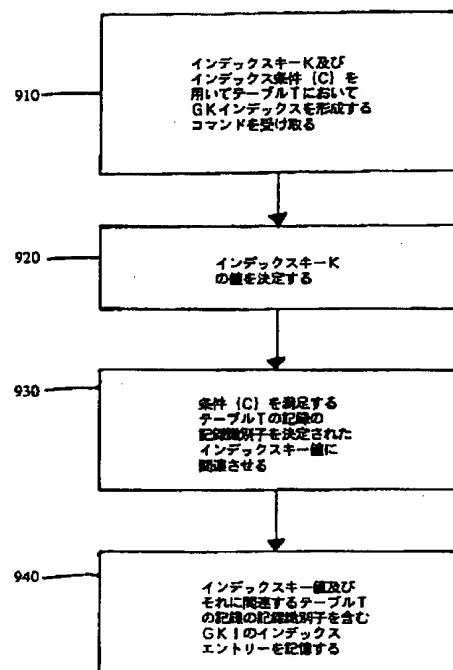
(74) 代理人 弁理士 中村 稔 (外7名)

(54) 【発明の名称】 一般化されたキーインデックス

(57) 【要約】

【課題】 リレーショナルデータベースシステムにおける新規なインデックスを提供する。

【解決手段】 一般化されたキーインデックスは、リレーショナルデータベースの第1テーブルをインデックスキー及びインデックス条件を用いてインデックスできるようにし、インデックスキー及びインデックス条件のいずれか又は両方がデータベースの多数のテーブルを参照してもよいし、或いは第1テーブル以外の1つ以上のテーブルのフィールドを用いた表現であってもよい。一般化されたキーインデックスは、一般化されたインデックスキー値を第1テーブルの記録の記録識別子に直接関連付け、効率的な記憶及びデータ検索を行えるようにする。



【特許請求の範囲】

【請求項1】 インデックスキー及びインデックス条件に基づきマルチテーブルのリレーショナルデータベースにおける第1テーブルのインデックスを形成するためのコンピュータ実施方法において、

インデックスキー値を決定し、インデックスキーが第1テーブル以外のテーブルを参照する場合には、他のテーブルからのデータを検索して、インデックスキー値を決定することを含み、

各々の決定されたインデックスキー値を、インデックス条件を満足する第1テーブルの記録の記録識別子と関連させ、インデックス条件が第1テーブル以外のテーブルを参照する場合には、他のテーブルからのデータを検索して、関連する記録識別子を決定することを含み、そしてインデックスキー値及び関連する記録識別子をインデックスに記憶する、という段階を備えたことを特徴とする方法。

【請求項2】 上記インデックスキー値は、少なくとも第2テーブル及び第3テーブルを参照する請求項1に記載の方法。

【請求項3】 上記インデックス条件は、少なくとも第2テーブル及び第3テーブルを参照する請求項1に記載の方法。

【請求項4】 上記インデックスキーは、少なくとも第2テーブルを参照する表現である請求項1に記載の方法。

【請求項5】 上記インデックスキー表現は、更に、第3テーブルを参照する請求項4に記載の方法。

【請求項6】 インデックスキーに基づきマルチテーブルのリレーショナルデータベースにおける第1テーブルのインデックスを形成するためのコンピュータ実施方法において、

インデックスキー値を決定し、インデックスキーは、少なくとも第2テーブルを参照する表現を含み、各々の決定されたキー値を第1テーブルの記録の記録識別子と関連させ、そしてインデックスキー値及び関連する記録識別子をインデックスに記憶する、という段階を備えたことを特徴とする方法。

【請求項7】 上記インデックスキー表現は、更に、第3テーブルを参照する請求項6に記載の方法。

【請求項8】 マルチテーブルのリレーショナルデータベースにおける問合せを処理するためのインデックスの適用性を決定するコンピュータ実施方法であって、上記インデックスは、インデックスキー及びインデックス条件に基づいてインデックステーブルをインデックスし、上記問合せは、1組の問合せテーブルを参照する問合せ条件に基づくものであり、そして各々のテーブルは、マルチテーブルのリレーショナルデータベースのテーブルであり、上記方法は、インデックステーブルが問合せテーブルの1つであるか

どうか決定し、

インデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下であるかを決定し、そしてインデックステーブルが問合せテーブルの1つでありそしてインデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下である場合には、問合せを処理するのにインデックスを適用できると決定する、という段階を備えたことを特徴とする方法。

【請求項9】 インデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下であるかを決定する上記段階は、各インデックス条件が少なくとも1つの問合せ条件により満足されるかどうか決定することを含む請求項8に記載の方法。

【請求項10】 リレーショナルデータベースの1組の問合せテーブルに対して問合せを処理するコンピュータ実施方法において、

インデックスキー値を有するインデックスエントリーと、上記1組の問合せテーブルにおける第1問合せテーブルの記録の関連記録識別子とを含むインデックスを検索し、上記インデックスエントリーと、その関連記録識別子との関係は、少なくとも上記1組の問合せテーブルにおける第2の問合せテーブル及び第3の問合せテーブルに基づくものであり、

上記問合せに基づいてインデックスキー値を決定し、その決定されたインデックスキー値に対応するインデックスエントリーを決定し、そしてその決定されたインデックスエントリーの関連記録識別子に基づいて第1問合せテーブルの記録を検索する、という段階を備えたことを特徴とする方法。

【請求項11】 リレーショナルデータベースの1組の問合せテーブルに対して問合せを処理するコンピュータ実施方法において、

インデックスキー値を有するインデックスエントリーと、上記1組の問合せテーブルにおける第1問合せテーブルの記録の関連記録識別子とを含むインデックスを検索し、上記インデックスキー値は、少なくとも上記1組の問合せテーブルにおける第2の問合せテーブルを参照する表現に基づくものであり、

上記問合せに基づいてインデックスキー値を決定し、その決定されたインデックスキー値に対応するインデックスエントリーを選択し、そしてその選択されたインデックスエントリーに基づいて第1問合せテーブルの記録を検索する、という段階を備えたことを特徴とする方法。

【請求項12】 プログラム可能なプロセッサがインデックスキー及びインデックス条件に基づいてマルチテーブルのリレーショナルデータベースの第1テーブルの記録をオーダーできるようにプロセッサ読み取り可能な媒体に記憶されたインデックスにおいて、インデックスキー値と、

インデックス条件を満足する第1テーブルの記録に対する関連記録識別子とを備え、

上記インデックスキー及びインデックス条件の少なくとも1つは、マルチテーブルのリレーショナルデータベースの第2及び第3テーブルを参照することを特徴とするインデックス。

【請求項13】 プログラム可能なプロセッサがインデックスキーに基づいてマルチテーブルのリレーショナルデータベースの第1テーブルの記録をオーダーできるようにプロセッサ読み取り可能な媒体に記憶されたインデックスにおいて、

インデックスキー値と、

第1テーブルの記録に対する関連記録識別子とを備え、上記インデックスキーは、少なくともマルチテーブルのリレーショナルデータベースの第2テーブルを参照する表現であることを特徴とするインデックス。

【請求項14】 上記インデックスキーの表現は、更に、第3テーブルを参照する請求項13に記載のインデックス。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、一般に、リレーショナルデータベースシステムに係り、より詳細には、リレーショナルデータベースシステムにおけるインデックス動作に係る。

【0002】

【従来の技術】 ユーザは、リレーショナルデータベースをテーブルの集合体と理解しており、各テーブルは、項目及び項目属性のデータを行及び列に配列するもので、テーブルの行は、項目（記録と称する）に対応し、そしてテーブルの列は、項目の属性（フィールドと称する）に対応する。テーブルの記録は、一般的に、その記録が記憶されたページ及びページ内のオフセットを指示する記録識別子（RID）により識別することができる。

【0003】 図1ないし3は、リレーショナルデータベースにおけるテーブルの例を示す。テーブルに記憶されたデータは、図4ないし6に示されたようなインデックスを用いてアクセスすることができ、これらインデックスは、テーブル記録の論理的なオーダー（順序）を与えることによりデータをアクセスするのに役立つ。図4は、顧客名410aをインデックスキーとして使用する図1のCUSTテーブル100における従来のインデックス400を示す。インデックス400は、顧客名120によりCUSTテーブル100の記録のアルファベット式オーダーを与え、各名前410aを、cust_nameフィールド120に顧客名を有するテーブル記録の記録識別子410bと関連付ける。

【0004】 Smithという名前の顧客の電話番号の問合せは、通常は、関連記録識別子440bを用いてキー値「Smith」440aを有するエントリーのイン

デックス400をサーチしてCUSTテーブル100における記録を検索し、そしてその検索された記録のcust_phoneフィールド140のデータを抽出することにより処理される。インデックス400は、顧客名によりオーダーされるので、インデックスサーチは、インデックスなしに要求されるCUSTテーブル100の直線的なサーチよりも高速である。インデックスは、Bツリー及びハッシュ機能のような種々の方法を用いて実施される。

【0005】 インデックス400で例示されたように、インデックスエントリーは、一般的に、インデックスキー値、及びテーブル記録の記録識別子の2つの属性を有している。インデックスキーは、一般に、テーブルからのフィールド又はフィールドの組み合わせであり、そして各インデックスエントリーは、キー値をテーブル記録の記録識別子と関連付ける。

【0006】 従来のインデックスは、テーブルフィールドに基づくキー値を、その同じテーブルの記録の記録識別子に関連付ける単一テーブルである。図4のインデックス400は、単一テーブルインデックスの一例である。単一テーブルインデックスは、種々の形式が存在する。図7は、テーブル記録の1つ以上のフィールドを用いた表現としてインデックスキーを定義することにより形成された「仮想列」インデックスの一例を示す。例えば、割引を含む価格により図2のオーダーテーブル（ORDテーブル）の記録をオーダーするために、インデックスキー（ORD_price-ORD_discount）を用いて仮想列インデックスが形成される。

【0007】 別の形式の単一テーブルインデックスは、指定の条件を満足するテーブル記録のみをインデックスする「部分インデックス」である。例えば、図8は、ORD_dateの値を、少なくとも\$50,000.00の価格を有するテーブル記録の記録識別子のみに関連付けるインデックスを示している。

【0008】 リレーショナルデータベースシステムにおいて、インデックスは、マルチテーブルであって、そのインデックスキーの第1テーブルからのフィールドを使用しそしてインデックスキー値を第2テーブルの記録の記録識別子に関連付けるものであってもよい。マルチテーブルインデックスは、一般に、連結インデックスと称する。

【0009】 図4ないし6は、CUSTテーブル100からの名前フィールド120をインデックスキー（CUST.cust_name）として使用してITEMテーブル300（図3）に形成された顧客名に基づくインデックス記録を検索するためのインデックスを実施を示す。図示されたように、このインデックスは、3つのレベルを有する。第1のレベル（インデックス400、図4）は、CUST.cust_name410aの値をCUSTテーブル410bの記録の記録識別子に関連付

け、第2のレベル（インデックス500、図5）は、CUST. cust_no510aの値を、そのORD. ord_no610aの記録の記録識別子に関連付け、そして第3のレベル（インデックス600、図6）は、ORD. ord_no610aの値を、ITEM. ord_no610bの記録の記録識別子に関連付ける。

【0010】インデックス400ないし600は、顧客名に対応する項目記録を検索する上で役立つ。例えば、顧客Smithの項目記録を検索するためには、インデックス400は、Smith(002)に対応するCUST記録のCUST. cust_no(1002)を検索するのに使用され、インデックス500は、CUST. cust_no(1002)に対応するORD記録(101、102)のORD. ord_no(002、003)を検索するのに使用され、そして最後にインデックス600は、ORD. ord_no(002、003)に対応するITEM記録(201、202、203)を検索するのに使用される。

【0011】

【発明が解決しようとする課題】このようなマルチテーブルインデックスは、リレーショナルデータベースシステムのデータを簡単にアクセスするが、リレーショナルデータベースの従来のインデックスには制約がある。例えば、第1テーブルのフィールドのキー値を第2テーブルの記録の記録識別子に関連付ける従来のインデックスは、一般に、図4ないし6に示すように、2つ以上のレベルを有する。

【0012】マルチテーブルインデックスにおいて多数のレベルを有する結果として、このようなインデックスは、単一テーブルインデックスよりも多量のメモリスペースを必要とする。更に、多数のレベルを有する結果として、問合せを処理するのに多量の処理が必要となる。例えば、顧客名Smithに対応する項目記録を検索する場合に、多数の処理段階が必要となる。

【0013】

【課題を解決するための手段】一般に、本発明は、その1つの特徴において、インデックスキー及びインデックス条件に基づきマルチテーブルのリレーショナルデータベースにおける第1テーブルのインデックスを形成するためのコンピュータ実施方法であって、インデックスキー値を決定し、インデックスキーが第1テーブル以外のテーブルを参照する場合には、他のテーブルからのデータを検索してインデックスキー値を決定することを含み；更に、各々の決定されたインデックスキー値を、インデックス条件を満足する第1テーブルの記録の記録識別子と関連させ、インデックス条件が第1テーブル以外のテーブルを参照する場合には、他のテーブルからのデータを検索して、関連する記録識別子を決定すること

を含み；そしてインデックスキー値及び関連する記録識別子をインデックスに記憶するという段階を備えた方法を提供する。

【0014】本発明の実施形態は、次の特徴の1つ以上を含む。インデックスキーは、少なくとも第2テーブル及び第3テーブルを参照する。インデックス条件は、少なくとも第2テーブル及び第3テーブルを参照する。インデックスキーは、少なくとも第2テーブルを参照する表現である。

【0015】本発明は、その別の特徴において、インデックスキーに基づきマルチテーブルのリレーショナルデータベースにおける第1テーブルのインデックスを形成するためのコンピュータ実施方法であって、インデックスキー値を決定し、インデックスキーは、少なくとも第2テーブルを参照する表現を含み；各々の決定されたキー値を第1テーブルの記録の記録識別子と関連させ；そしてインデックスキー値及び関連する記録識別子をインデックスに記憶するという段階を備えた方法も提供する。

【0016】本発明は、その更に別の特徴において、マルチテーブルのリレーショナルデータベースにおける問合せを処理するためのインデックスの適用性を決定するコンピュータ実施方法であって、上記インデックスは、インデックスキー及びインデックス条件に基づいてインデックステーブルをインデックスし、上記問合せは、1組の問合せテーブルを参照する問合せ条件に基づくものであり、そして各々のテーブルは、マルチテーブルのリレーショナルデータベースのテーブルであるような方法も提供する。この方法は、インデックステーブルが問合せテーブルの1つであるかどうか決定し；インデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下であるかを決定し；そしてインデックステーブルが問合せテーブルの1つでありそしてインデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下である場合には、問合せを処理するのにインデックスを適用できると決定するという段階を備えている。

【0017】本発明の実施形態は、インデックス条件が問合せ条件に比して等しく限定的であるか又はそれ以下であるかを決定する段階が、各インデックス条件が少なくとも1つの問合せ条件により満足されるかどうか決定することを含むような方法も包含する。

【0018】本発明は、その更に別の特徴において、リレーショナルデータベースの1組の問合せテーブルに対して問合せを処理するコンピュータ実施方法であって、インデックスキー値を有するインデックスエントリーと、上記1組の問合せテーブルにおける第1問合せテーブルの記録の関連記録識別子とを含むインデックスを検索し、上記インデックスエントリーと、その関連記録識別子との関係は、少なくとも上記1組の問合せテーブルにおける第2の問合せテーブル及び第3の問合せテー

ルに基づくものであり；上記問合せに基づいてインデックスキー値を決定し；その決定されたインデックスキー値に対応するインデックスエントリーを決定し；そしてその決定されたインデックスエントリーの関連記録識別子に基づいて第1問合せテーブルの記録を検索するという段階を備えた方法も提供する。

【0019】本発明は、その更に別の特徴において、リレーショナルデータベースの1組の問合せテーブルに対して問合せを処理するコンピュータ実施方法であって、インデックスキー値を有するインデックスエントリーと、上記1組の問合せテーブルにおける第1問合せテーブルの記録の関連記録識別子を含むインデックスを検索し、上記インデックスキー値は、少なくとも上記1組の問合せテーブルにおける第2の問合せテーブルを参照する表現に基づくものであり；上記問合せに基づいてインデックスキー値を決定し；その決定されたインデックスキー値に対応するインデックスエントリーを選択し；そしてその選択されたインデックスエントリーに基づいて第1問合せテーブルの記録を検索するという段階を備えた方法も提供する。

【0020】本発明は、その更に別の特徴において、プログラム可能なプロセッサがインデックスキー及びインデックス条件に基づいてマルチテーブルのリレーショナルデータベースの第1テーブルの記録をオーダーできるようにプロセッサ読み取り可能な媒体に記憶されたインデックスであって、インデックスキー値と、インデックス条件を満足する第1テーブルの記録に対する関連記録識別子とを備え、上記インデックスキー及びインデックス条件の少なくとも1つは、マルチテーブルのリレーショナルデータベースの第2及び第3テーブルを参照するようなインデックスも提供する。

【0021】本発明は、その更に別の特徴において、プログラム可能なプロセッサがインデックスキーに基づいてマルチテーブルのリレーショナルデータベースの第1テーブルの記録をオーダーできるようにプロセッサ読み取り可能な媒体に記憶されたインデックスであって、インデックスキー値と、第1テーブルの記録に対する関連記録識別子とを備え、上記インデックスキーは、少なくともマルチテーブルのリレーショナルデータベースの第2テーブルを参照する表現であるようなインデックスも提供する。

【0022】本発明の効果は、次の通りである。この一般化されたキーインデックスシステムは、単一レベルで実施することができ、効率的な記憶を達成すると共に、効率的な問合せ処理を可能にする。又、一般化されたキーインデックスは、例えば、2つ以上のテーブルのフィールドを用いた表現を含む種々のキーインデックスに対してインデックスすることができる。

【0023】更に、この一般化されたキーインデックスは、従来の単一テーブルインデックスを含む種々のイン

デックスフォーマットを包含するに充分なほどの融通性がある。リレーショナルデータベースシステムは、一般化されたキーインデックス及び従来のインデックスの両方を形成しそして取り扱うように容易に適用できる。

【0024】本発明の更に別の目的及び効果は、以下の説明から、又は本発明の実施により明らかとなろう。本発明の目的及び効果は、特許請求の範囲に特に指摘する手段及びその組み合わせにより実現することができよう。

【0025】

【発明の実施の形態】以下、添付図面を参照して、本発明の好ましい実施形態を詳細に説明する。本発明は、リレーショナルデータベースシステムにおけるインデックスの一般化された使用を提供する。一般化されたキーインデックス（GKインデックス）は、仮想列インデックス及び部分インデックスのような単一テーブルインデックス、並びにマルチテーブルインデックスを含む従来のインデックスを包含し、そして従来のインデックスに使用された異なる技術を単一のインデックスに結合できるようにする。更に、GKインデックスは、効率的に記憶されるカスタマイズされたインデックスを形成しそして効率的な問合せ処理を行えるようにする改良された能力を提供する。

【0026】1つの実施形態において、GKインデックスは、次の一般的フォーマットを有するユーザコマンドにより形成される。

```
CREATE GK-INDEX    (index-name)
ON                  (table-name)
(SELECT AS KEY      (key-list)
FROM                (other-table-list)
WHERE                (condition-list))
```

但し、パラメータは、次の通りである。

index-name: 形成されているインデックスを識別する。

table-name: 記録がインデックスされているテーブルを識別する。

【0027】key-list: リレーショナルデータベースの1つ以上のテーブルからの1つ以上のフィールドを含む。

other-table-list: テーブル名に対応するテーブル以外のテーブルであって、キーリスト又は条件リストのいずれかで参照されたフィールドを有する各テーブルを含む。形成されているインデックスが単一テーブルインデックスである場合には、このリストは、空となる。

condition-list: 満足されるべき0以上の条件（表現）を含む。このリストは空である。

【0028】1つの実施形態は、問合せを処理しそしてその結果を記憶するようなステートメントからGKインデックスを形成する。例えば、GKインデックスを形成する一般的フォーマットは、問合せとして表される。

```

SELECT      <:key-list>:, <:table-name>:. RID
FROM        <:other-table-list>:, <:table-name>;
WHERE       <:condition-list>;
ORDER BY    <:order-list>;

```

但し、<:table-name>:. RID は、テーブル記録を参照し、そして<:order-list>;は、一般に、キーリストにおいてオーダーされたフィールドを反映する。

【0029】GKインデックスは、仮想列インデックス及び部分インデックスを含む従来の単一テーブルインデックスを形成するのに使用することもできる。上記フォーマットを用いた実施においては、単一テーブルGKインデックスは、次のようなステートメントにより形成される。

```

CREATE GK-INDEX:  (index-name)
ON                (table-name)
                (SELECT AS KEY (key-list)
                WHERE          (condition-list))

```

これは、次のように問合せとして表される。

```

SELECT      <:key-list>:, <:table-name>:. RID
FROM        <:table-name>;
WHERE       <:condition-list>;
ORDER BY    <:order-list>;

```

【0030】従って、例えば、名前によりCUSTテーブルをインデックスする図4を参照して述べた標準的な1テーブルインデックスは、次のようなステートメントによって形成される。

```

CREATE GK-INDEX    Cust__Name__Index
ON                 Cust
                (SELECT AS KEY CUST.cust __name)

```

これは、次のように問合せとして表される。

```

SELECT      CUST.cust __name, CUST.RID
FROM        CUST
ORDER BY    CUST.cust __name

```

【0031】図7を参照して上記した仮想列インデックス700は、次のステートメントによって形成される。

```

CREATE GK-INDEX    VC__Index
ON                 ORD
                (SELECT AS KEY (ORD.price - ORD.discount)

```

これは、次のように問合せとして表される。

```

SELECT      (ORD.price - ORD.discount)
            ORD.RID
FROM        ORD
ORDERED BY  (ORD.price - ORD.discount)

```

【0032】同様に、図8を参照して述べた部分インデックス800は、次のように形成される。

```

CREATE GK-INDEX    CUST__item
ON                 ITEM

```

```

CREATE GK-INDEX    Partial __Index
ON                 ORD
                (SELECT AS KEY ORD.date
                WHERE          ORD.price ≥ 50,000.00)

```

これは、次のように問合せとして表される。

```

SELECT      ORD.date, ORD.RID
FROM        ORD
WHERE       ORD.price ≥ 50,000.00
ORDER BY    ORD.date

```

【0033】上記のように、GKインデックスは、従来の単一テーブルインデックスを包含する。又、GKインデックスは、マルチテーブルであってもよい。マルチテーブルGKインデックスに対応する問合せを処理する際には、GKインデックスキー値と、インデックスされたテーブルの記録の記録識別子との最終的な関連を決定するのに中間テーブルが参照されるが、GKインデックスは、最終的な結果を記憶する。

【0034】問合せ処理方法は、広範囲に研究されており、「Create GK Index (GKインデックスを形成する)」コマンドに対応する問合せを処理するのに種々の方法を使用することができる。

【0035】図9は、GKインデックスを形成する方法を示すフローチャートである。ステップ910において、インデックスキーK (<:key-list>として上記した)及びインデックス条件[C] (<:condition-list>として上記した)を使用して、テーブルT (<:table-name>として上記した)にGKインデックスGKIを形成するコマンドを受け取った後に、ステップ920においてインデックスキーKの値が決定される。ステップ930において、各インデックスキー値に対し、条件[C]を満足するテーブルTの関連記録識別子が決定される。ステップ940において、インデックスキー値及びそれらの関連記録識別子が記憶される。

【0036】インデックスキー値に対してテーブルTの記録の関連記録識別子を決定するときには、テーブルTではなく、中間テーブル(<:other-table-list>のテーブルの1つ)を参照することができる。例えば、<:other-table-list>のテーブルがインデックスキーにより又は[C]の条件において参照される場合には、他のテーブルからのデータを検索する必要がある。しかしながら、中間テーブルが参照されるかどうかに関わりなく、インデックスは、インデックスキー値を、インデックスされたテーブルTの記録の関連記録識別子に直接関連させるマッピングの結果を記憶する。

【0037】図10は、顧客名を項目記録の記録識別子に関連付けて、図4ないし6のインデックスと同じ機能を実行するGKインデックス1000を示す。次のようなコマンドの結果、


```

(SELECT AS KEY  CUST.cust __name
FROM            CUST
WHERE           CUST.cust __no = ORD.cust __no AND
                ORD.ord __no = ITEM.ord __no)

```

それに対応する問合せ

```

SELECT          CUST.cust __name, ITEM.RID
FROM            CUST, ORD, ITEM
WHERE           CUST.cust __no = ORD.cust __no AND
                ORD.ord __no = ITEM.ord __no
ORDER BY       CUST.cust __name

```

が処理され、図10のGKインデックス1000を形成する。

【0038】図4のインデックス400と同様に、GKインデックス1000は、顧客名を項目記録の記録インデックスに関連付ける。しかしながら、図示されたように、GKインデックス1000は、CUSTテーブルからの顧客名の値をITEMテーブルの記録の記録識別子に直接関連付ける単一のレベルしかもたず、ORDテーブルへの参照を排除する。

【0039】図4ないし6のインデックスのような従来のマルチテーブルインデックスは、キー値をテーブル記録の記録識別子に関連付けるために中間記録を参照する多数のレベルを有している。多数のレベルは、多量のメモリを必要としそしてアクセス時間を遅くするだけでなく、形成することのできるインデックスの形式にある程

```

CREATE GK-INDEX cust_expenditure
ON              CUST
                (SELECT AS KEY ((ORD.price - ORD.discount) *
                                (1 + ORD.tax))
FROM            ORD
WHERE           ORD.cust __no = CUST.cust __no)

```

これは、次のように処理される。

```

SELECT          ((ORD.price - ORD.discount) *
                (1 + ORD.tax)), CUST.RID
FROM            ORD, CUST
WHERE           ORD.cust __no = CUST.cust __no

```

【0042】又、GKインデックスキーは、2つ以上のテーブルからのフィールドに形成することもできる。例えば、次のコマンドは、顧客名及びオーダー日付の組み

```

CREATE GK-INDEX cust_date_item
ON              ITEM
                (SELECT AS KEY (CUST.cust __name,
                                ORD.date)
FROM            CUST, ORD
WHERE           CUST.cust __no = ORD.cust __no AND
                ORD.ord __no = ITEM.ord __no)

```

これは、次により形成される。

```

SELECT          CUST.cust __name,
                ORD.date), ITEM.RID
FROM            CUST, ORD, ITEM

```

度の制約を課する。例えば、従来のマルチレベルマルチテーブルインデックス動作は、インデックスキーが2つ以上のテーブルのフィールドに基づくことを許さない。

【0040】これに対して、GKインデックスは、GKインデックスキー値をテーブル記録の記録識別子に直接関連させる単一レベルしかもたないので、ユーザは、関連をいかに定めるかを決定する上で著しい融通性をもつ。例えば、GKインデックスキーは、第2テーブルの1つ以上のフィールドを用いる表現を含むインデックスキーを使用して第1テーブルに形成することができる。図1ないし3のテーブルを参照すれば、特定の全価格を有する単一オーダー購入を行った顧客を決定するために、図11のGKインデックスは、次のように形成される。

【0041】

合わせを、そのオーダーで購入した項目の記録識別子に関連付ける図12のGKインデックス1200を形成する。

【0043】

```

WHERE          CUST.cust __no = ORD.cust __no AND
                ORD.ord __no = ITEM.ord __no
ORDER BY       CUST.cust __name, ORD.date

```

【0044】これらの例は、ユーザの特定のニーズに対してカスタマイズされたインデックスを形成するための一般化されたキーインデックス動作の融通性を示している。

【0045】GKインデックスを形成するために参照されたテーブルにおいてデータが追加され、削除され又は変更された後にGKインデックスの精度の継続を確保するために、GKインデックスは、データ変更が生じた後にそのエントリーが依然として正確であることを照合するために更新されねばならない。

【0046】種々の更新方法が存在し、GKインデックスを更新するのに適用できる。1つの実施形態では、参照されたフィールドのいずれかが変更されたときにGKインデックスが再構成されるか、又はその変更がGKインデックスに適さない場合にはGKインデックスが削除される（例えば、不適合のデータ形式に対しフィールドが削除されるか又は変更される）。この方法は、データが頻繁に更新されないデータウェアハウス（倉庫）環境に特に適している。

【0047】データベースシステムは、GKインデックスを使用し、GKインデックス内のGKインデックスキー値を見出して、関連テーブル記録への関連記録識別子に従うことにより、データをアクセスする。例えば、Smithという名前の顧客に対する項目記録を見つけるために図10のGKインデックス1000を使用する場合に、Smithのエントリー1030aに対してGKインデックス1000が最初にサーチされ、そしてそれに対応する記録識別子1030bにより項目記録が検索される。

【0048】インデックスを使用するために、リレーショナルデータベースシステムは、最初に、所与の問合せを処理するためにその特定のインデックスが適切に使用されるかどうか決定する。典型的に、データベースシステムは、所与の問合せを処理するのにどのインデックスが使用されるかを決定する役目を果たす問合せ最適化手段を有している。

【0049】一般に、問合せ最適化手段は、（1）インデックスを問合せに適用できるかどうか及び（2）問合せを処理するようにインデックスが最適に使用されるかどうかを決定することにより、所与の問合せに対し特定のインデックスを使用すべきかどうか決定する。例えば、次の問合せQについて考える。

```

SELECT          <:query-select-list>;
FROM            <:query-table-list>;

```

```

CREATE GK-INDEX cust_big __items
ON              ITEM
                (SELECT AS KEY CUST.cust __name

```

```

WHERE          <:query-condition-list>;

```

更に、次のように形成されるGKインデックスGKIについて考える。

```

CREATE GK-INDEX GKI
ON              T
                (SELECT AS KEY <:index-key-list>
FROM            <:index-other-table-list>
WHERE           <:index-condition-list>)

```

【0050】一般に、GKインデックスを問合せに適用できるためには、それがインデックスするテーブルが問合せにより参照されねばならない。図13は、適用性を決定する1つの実施形態におけるプロセスを示すフローチャートである。ステップ1310において、GKインデックスGKIによりインデックスされたテーブルTが<:query-table-list>のテーブルの1つでない場合には、GKインデックスGKIは問合せに適用できない（ステップ1315）。同様に、ステップ1320において、<:index-condition-list>が<:query-condition-list>より制約がある場合には、GKインデックスGKIは問合せに適用できない（ステップ1315）。さもなくば、GKインデックスGKIは、問合せに適用できる（ステップ1325）。

【0051】制約度の評価（ステップ1320）は、インデックスGKIの使用が不完全な結果を生じないように確保し、問合せにより要求される以上に制約のある条件でインデックスGKIが形成される場合には不完全な結果が生じる。1つの実施形態では、制約度の条件を満足するかどうかは、<:index-condition-list>の各条件が<:query-condition-list>の少なくとも1つの条件により満足されるかどうかを評価することにより決定される。

【0052】以下の例は、適用性の考え方を示すものである。図1ないし3のテーブルを用いて、次のインデックスが形成される。

（1）cust__items インデックス：図10に示すように、cust__items インデックス1000は、顧客名を、その顧客が購入した項目に対する項目記録の記録識別子と関連させる。

（2）cust_big __items インデックス：図14に示すように、cust_bigitems インデックス1400は、顧客名を、少なくとも\$50,000.00の価格のオーダーでその顧客が購入した項目に対する項目記録の記録識別子と関連させる。このインデックスは、次のように形成される。

```

FROM          CUST, ORD
WHERE          CUST.cust __no = ORD.cust __no AND
              ORD.ord __no = ITEM.ord.no AND
              ORD.price ≥ 50,000.00)

```

(3) cust_many_big_items インデックス: 図 15 くと 10 の量で購入された項目に対する項目記録の記録識別子と関連させる。このインデックスは、次のように示すように、custmany_big_items インデックス 1 録識別子と関連させる。このインデックスは、次のように形成される。

500 は、顧客名を、少なくとも \$50,000.00 に形成される。

の価格のオーダーでその顧客により購入されそして少な 【0053】

```

CREATE GK-INDEX cust_many_big_items
ON              ITEM
(SELECT AS KEY  CUST.cust __name
FROM            CUST, ORD
WHERE           CUST.cust __no = ORD.cust __no AND
              ORD.ord __no = ITEM.ord.no AND
              ORD.price ≥ 50,000.00) AND
              ITEM.quantity ≥ 10)

```

【0054】次の問合せは、図 1 ないし 3 のテーブルを 量を見出す。この問合せは、次のようにフォーマットされる。

含むデータベースに基づいて送られる。(1) Q1: 顧客 Smith によるオーダーにおいて項目番号及び項目

```

SELECT        ITEM.item __no, ITEM.quantity
FROM          CUST, ORD, ITEM
WHERE         CUST.cust __name = "Smith" AND
              CUST.cust __no = ORD.cust __no AND
              ORD.ord __no = ITEM.ord __no

```

(2) Q2: 顧客 Smith による少なくとも \$60,000.00 のオーダーにおいて項目番号及び項目量を見出す。この問合せは、次のようにフォーマットされる。

【0055】

```

SELECT        ITEM.item __no, ITEM.quantity
FROM          CUST, ORD, ITEM
WHERE         CUST.cust __name = "Smith" AND
              CUST.cust __no = ORD.cust __no AND
              ORD.ord __no = ITEM.ord __no AND
              ORD.price ≥ 60,000.00

```

(3) Q3: 顧客 Smith による少なくとも \$60,000.00 の少なくとも 25 の量のオーダーにおいて 購入した項目番号及び項目量を見出す。この問合せは、次のようにフォーマットされる。

```

SELECT        ITEM.item __no, ITEM.quantity
FROM          CUST, ORD, ITEM
WHERE         CUST.cust __name = "Smith" AND
              CUST.cust __no = ORD.cust __no AND
              ORD.ord __no = ITEM.ord __no AND
              ORD.price ≥ 60,000.00 and
              ITEM.quantity ≥ 25

```

【0056】問合せ Q1、Q2 及び Q3 に対するインデックス 1000、1400 及び 1500 の適応性は、図 16 に示されている。Q1 は、Smith により購入された全ての項目に関する情報を要求する。インデックス 1000 のみを適用できる。というのは、インデックス 1400 及び 1500 のインデックス条件は、問合せ条件よりも限定的であり、完全な結果を生じないからである。ORD.price ≥ 50,000 というインデックス 1400

の条件は、Q1 の条件により満足されない。従って、インデックス 1400 は、少なくとも \$50,000.00 のオーダーにおいて Smith が購入した項目しか指示しない。インデックス 1500 は、ORD.price ≥ 50,000 及び ITEM.quantity ≥ 10 の両方の条件を含み、そのいずれも Q2 によって満足されない。従って、インデックス 1500 は、少なくとも \$50,000.00 のオーダーにおいて Smith が購入した項目のみを指示

する。

【0057】この例も、問合せ条件よりも限定的な条件を有するインデックスは誤った結果を招くことを示している。即ち、Smithが購入した全ての項目を要求するQ1に対する適切な応答は、項目201、202、203に対応し、インデックス1400は、項目201及び202のみを生じ、インデックス1500は、項目201のみを生じる。

【0058】適用性の概念は、問合せQ2及びQ3によって更に示される。Q2は、少なくとも\$60,000.00のオーダーにおいてSmithが購入した項目を要求する。インデックス1400は、ORD.price \geq 50,000.00というその条件がORD.price \geq 60,000.00というQ2問合せ条件により満足されるので、Q2に適用することができる。インデックス1500は、ORD.price \geq 50,000 というその条件は満足されるが、ORD.quantity \geq 10 というその条件がQ2の条件によって満足されないので、適用できない。

【0059】Q3は、少なくとも\$60,000.00のオーダーにおいて25より大きな量でSmith氏が購入した項目を要求する。インデックス1400は、Q2に対して適用できるという同じ理由で適用できる。インデックス1500も、ITEM.quantity \geq 10 というその条件がITEM.quantity \geq 25 というQ3問合せの条件により満足されるので、適用できる。

【0060】インデックスが所与の問合せに適用できることを決定すると、問合せ最適化手段は、一般に、その問合せを処理するのにインデックスが最適に使用されるかどうか決定する。最適化方法は、問合せ最適化手段の間で異なるが、一般的に使用される方法は、P. G. セリンガー氏等の「リレーショナルデータベースシステムのためのアクセス仕様言語 (An Access Specification Language for a Relational Data Base System)」、19

```
SELECT  CUST.name, ITEM.quantity
FROM    CUST, ORD, ITEM
WHERE   CUST.cust_name = "Smith" AND
        CUST.cust_no = ORD.cust_no AND
        ORD.ord_no = ITEM.ord_no AND
        ITEM.quantity  $\geq$  25
```

【0064】テーブルセットは、{CUST, ORD, ITEM}であり、プランは、次のような考えられる順列を含む。

- (1) {CUST, ORD, ITEM}
- (2) {CUST, ITEM, ORD}
- (3) {ORD, CUST, ITEM}
- (4) {ORD, ITEM, CUST}
- (5) {ITEM, CUST, ORD}
- (6) {ITEM, ORD, CUST}

【0065】図17のインデックス1700は、フィールドITEM.quantityをITEM.RIDと関連付け、そして条件ITEM.quantity \geq 25を処理するのに有用である。この

79 ACM SIGMOD INT'L CONF. ONMANAGEMENT OF DATA 23-34 (1979年5月/6月)に記載されたセリンガーの最適化アルゴリズムに基づいている。

【0061】セリンガーアルゴリズムの基本的な考え方は、所与の問合せを処理するための考えられる実行プランを列挙し、各考えられるプランのコストを推定し、そして最良のプラン（最もコストの安いプラン）を選択することである。従って、次の問合せを処理する場合、

```
SELECT  <:query-select-list>;
FROM    <:query-table-list>;
WHERE   <:query-condition-list>;
```

<:query-table-list>をセット {T1, T2, ..., Tn} とすれば、<:query-table-list>の全てのテーブルを接合することが必要となる。1つの考えられるセリンガー最適化手段は、問合せテーブルセットのテーブルの全ての順列に対するプランを形成し、そしてその順列の第1テーブルでスタートしてそのテーブルを第2テーブルと接合するコストを推定し、そのコストを、上記第1接合の結果を第3テーブルに接合するコストの推定値に加算し、テーブルセットの全てのテーブルを接合し続けそして累積された推定コストを計算することにより、各順列に対するコストを推定する。又、総コストは、いかなる接合も必要としない単一テーブル条件を組み込むコストも考慮する。このプロセスが各々の考えられる順列に対して繰り返される。

【0062】インデックスは、プランのコストを減少することができる。セリンガー最適化手段は、一般に、インデックスを用いて、テーブルを接合するコストを減少するか、又は単一テーブルに対して条件を決定するコストを減少する。

【0063】図1ないし3のテーブル100、200、300を参照し、次の問合せについて考える。

条件は、ITEMテーブルのみを参照し、それ故、他のテーブルを予め接合する必要はない。従って、インデックス1700は、ITEMテーブルが追加されるときに上記プラン(1)ないし(6)の各々に使用することができる。

【0066】図18のインデックス1800は、フィールドITEM.ord_noをITEM.RIDと関連付け、そして条件ORD.ord_no = ITEM.ord_noを遂行して、ITEMテーブルを共通のord_noフィールドによりORDテーブルに接合するのに有用である。しかしながら、この条件はORDテーブルを参照するので、インデックス1800

は、ORDテーブルが既に接合された場合にITEMテーブルを接合することのみに使用される。従って、インデックス1800は、プラン(1)、(3)及び(4)のみに使用される。

【0067】GKインデックスをいつ使用するかを決定する種々の方法を用いて、GKインデックスをこのアルゴリズムに組み込むようにセリンガー最適化手段を適応させることができる。例えば、1つの実施形態において、従来の単一テーブルインデックスと同様に、テーブルTにおける単一テーブルGKインデックスは、そのテーブルが接合されるときに、常に考慮される。マルチテーブルGKインデックスは、キーリストに参照された全てのテーブルが既に接合されたときに、考慮される。

【0068】セリンガーアルゴリズム以外のアルゴリズムをベースとする問合せ最適化手段も、GKインデックスを処理するのに同様に適応できるが、特定の異なる変更が行われる。

【0069】本発明の特許請求の範囲内において他の実施形態も考えられる。例えば、他の実施形態は、異なる更新方法を使用してもよい。ある実施形態は、その基礎となるテーブルが変更されるときにGKインデックスを削除するのではなく、インデックスを更新又は再計算する。

【0070】他の考えられる変更は、異なる形式のデータ構造を使用して、bツリー及びビットマップインデックスを含むGKキーインデックスを表すことを含む。

【0071】本発明は、ソフトウェアでの実施について説明したが、ハードウェア又はソフトウェア或いはその両方の組み合わせで実施することもできる。本発明は、プロセッサ、データ記憶システム、入力装置及び出力装置を備えたプログラム可能な処理システムにおいて実行されるソフトウェアプログラムで実施されるのが好ましい。

【0072】図19は、CPUバス1940により接続されたCPU1910、RAM1920及びI/Oコントローラ1930を含むプログラム可能な処理システム1900を示している。I/Oコントローラ1930は、I/Oバス1950によりキーボード1960のような入力装置、及びディスプレイ1970のような出力装置にも接続される。

【0073】以上、本発明を好ましい実施形態について説明したが、本発明は、これに限定されるものではなく、特許請求の範囲のみによって限定されるものとする。

【図面の簡単な説明】

【図1】リレーショナルデータベースのテーブルを示す図である。

【図2】リレーショナルデータベースのテーブルを示す図である。

【図3】リレーショナルデータベースのテーブルを示す図である。

【図4】インデックスを示す図である。

【図5】インデックスを示す図である。

【図6】インデックスを示す図である。

【図7】インデックスを示す図である。

【図8】インデックスを示す図である。

【図9】一般化されたキーインデックスを形成する方法のフローチャートである。

【図10】インデックスを示す図である。

【図11】インデックスを示す図である。

【図12】インデックスを示す図である。

【図13】一般化されたキーインデックスの適用性を決定する方法のフローチャートである。

【図14】インデックスを示す図である。

【図15】インデックスを示す図である。

【図16】所与の問合せに対するインデックスの適用性を示す図である。

【図17】インデックスを示す図である。

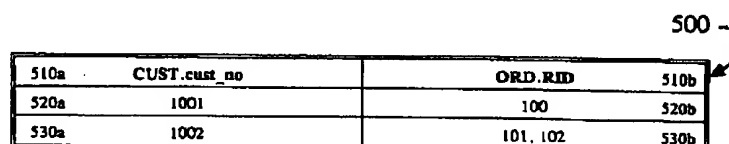
【図18】インデックスを示す図である。

【図19】プログラム可能な処理システムを示す図である。

【符号の説明】

1000、1400、1500 GKインデックス
1900 プログラム可能な処理システム
1910 CPU
1930 I/Oコントローラ
1940 CPUバス
1950 I/Oバス
1960 キーボード
1970 ディスプレイ

【図5】



510a	CUST.cust_no	ORD.RID	510b
520a	1001	100	520b
530a	1002	101, 102	530b

【図1】

100

110 RID	120 cust_name	130 cust_no	140 cust_phone
000	Johnson	1000	111-1111
001	Brown	1001	222-2222
002	Smith	1002	333-3333
003	Taylor	1003	444-4444

CUST table

従来例

【図2】

200

RID	ord_no	cust_no	price	discount	tax	date
100	001	1001	45,000.00	0.00	0.10	122095
101	002	1002	100,000.00	0.00	0.10	122095
102	003	1002	50,000.00	6,000.00	0.10	120195

ORD table

従来例

【図3】

300

RID	item_no	ord_no	quantity
200	111	001	45
201	222	002	30
202	333	002	5
203	444	003	50

ITEM table

従来例

【図4】

400

410a	CUST.cust_name	CUST.RID	410b
420a	Brown	001	420b
430a	Johnson	000	430b
440a	Smith	002	440b
450a	Taylor	003	450b

Cust_Name_index

従来例

【図6】

610a	ORD.ord_no	ITEM.RID	610b
620a	001	200	620b
630a	002	201, 202	630b
640a	003	203	640b

600

従来例

【図7】

(ORD.price - ORD.discount)	ORD.RID
44,000.00	102
45,000.00	100
100,000.00	101

700

VC_Index

従来例

【図8】

ORD.date	ORD.RID
120195	102
122095	101

800

Partial_Index

従来例

【図10】

1010a	CUST.cust_name	ITEM.RID	1010b
1020a	Brown	200	1020b
1030a	Smith	201, 202, 203	1030b

1000

cust_items index

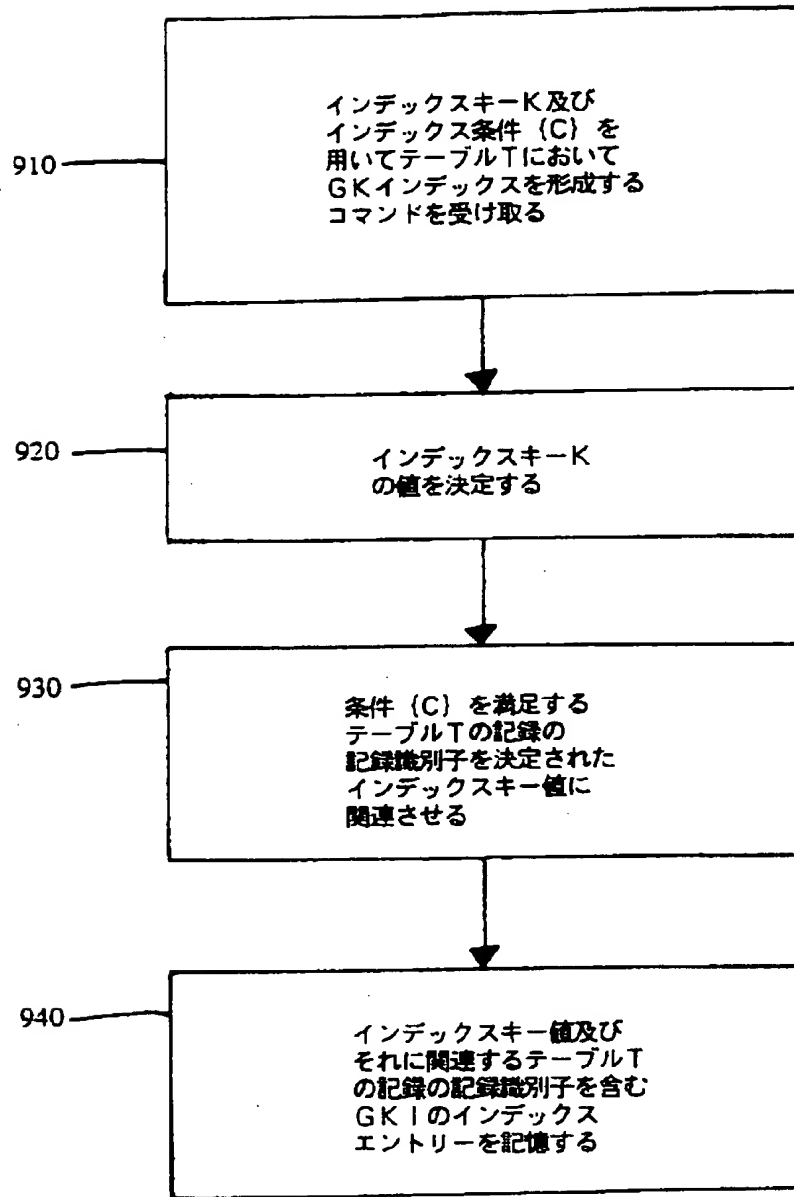
【図11】

(ORD.price - ORD.discount) * (1 + ORD.tax)	CUST.RID
48,400.00	002
49,500.00	001
110,000.00	002

1100

cust_expenditure index

【図 9】



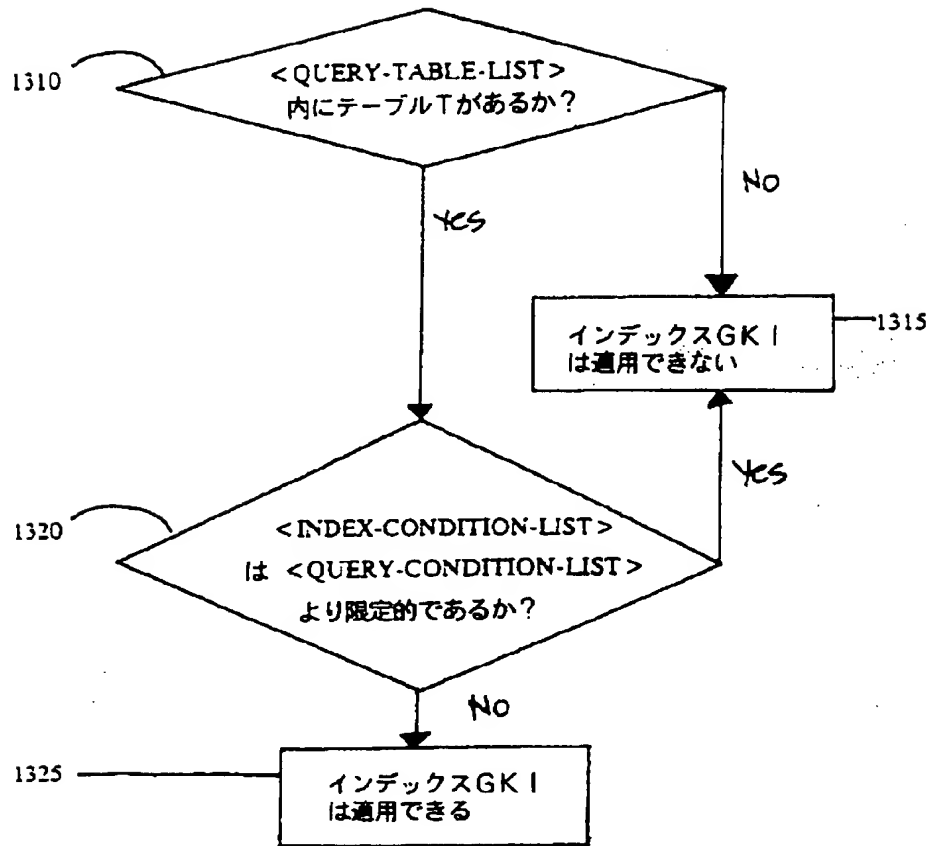
【図 12】

1200

CUST.cust_name, ORD.date	ITEM.RID
Brown, 122095	200
Smith, 120195	203
Smith, 122095	201, 202

cust_date_item index

【図 13】



【図 14】

1400

CUST.cust_name	ITEM.RID
Smith	201, 202

cust_big_items index

【図 15】

1500

CUST.cust_name	ITEM.RID
Smith	201

cust_many_big_items index

【図16】

	cust_items (1000)	cust_big_items (1300)	cust_many_big_items (1400)
Q1	Yes	No	No
Q2	Yes	Yes	No
Q3	Yes	Yes	Yes

【図17】

ITEM.quantity	ITEM.RID
5	202
30	201
45	200
50	203

1700

【図18】

ITEM.ord_no	ITEM.RID
001	200
002	201, 202
003	203

1800

【図19】

